US 5,956,409

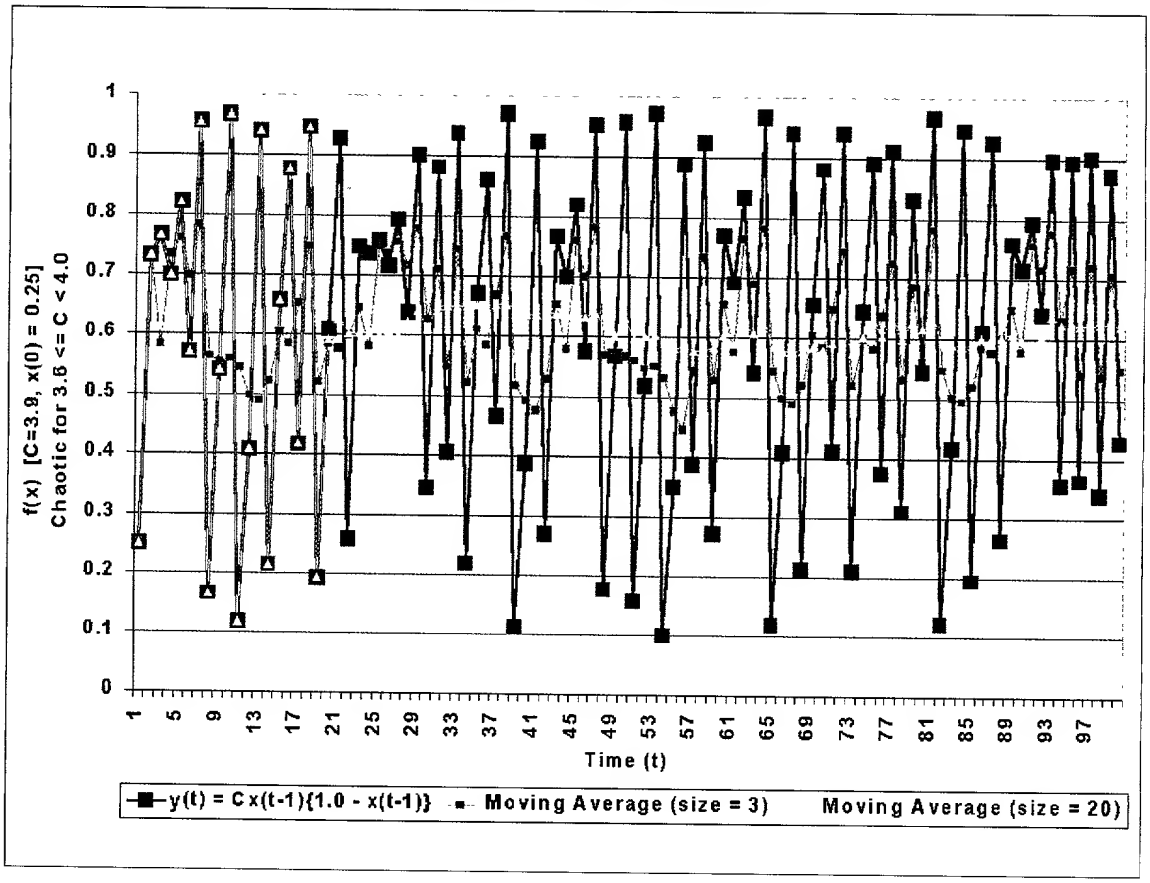


Fig. 1. Prior Art

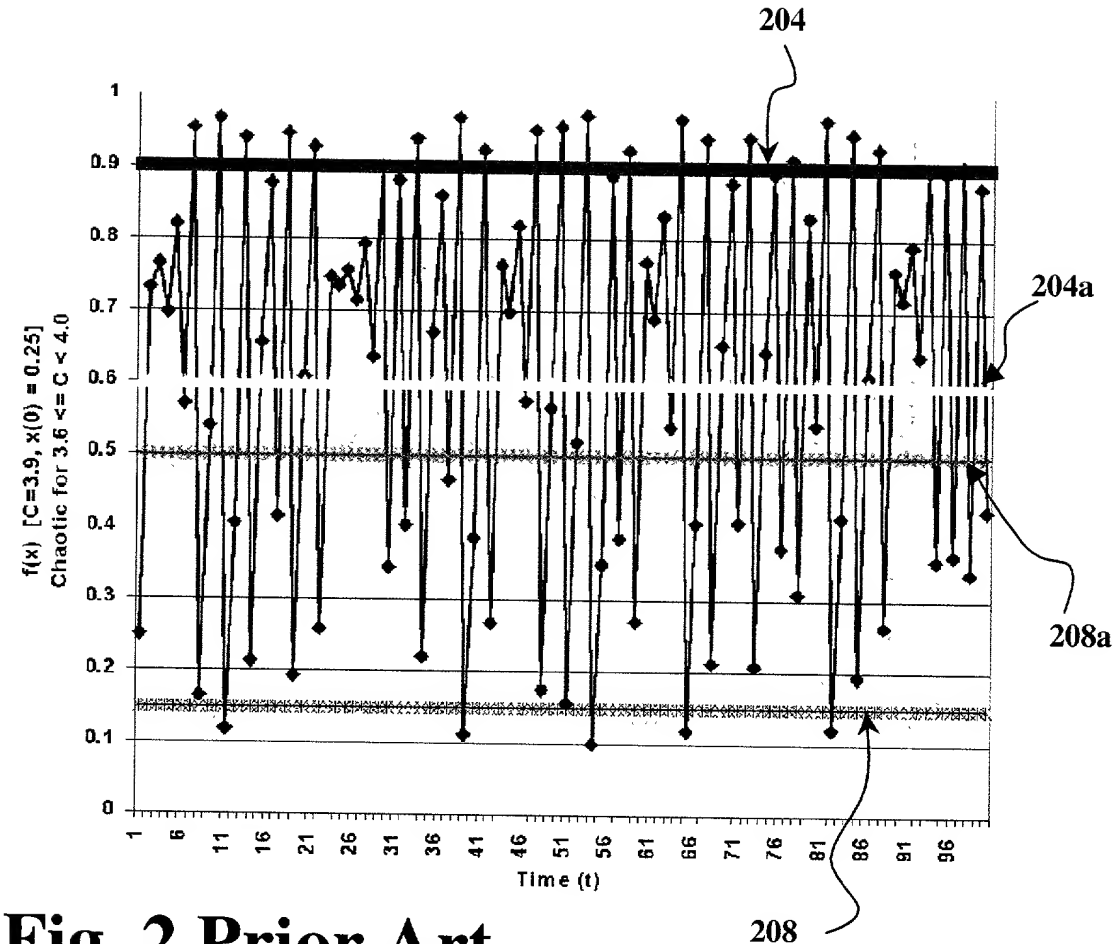


Fig. 2 Prior Art

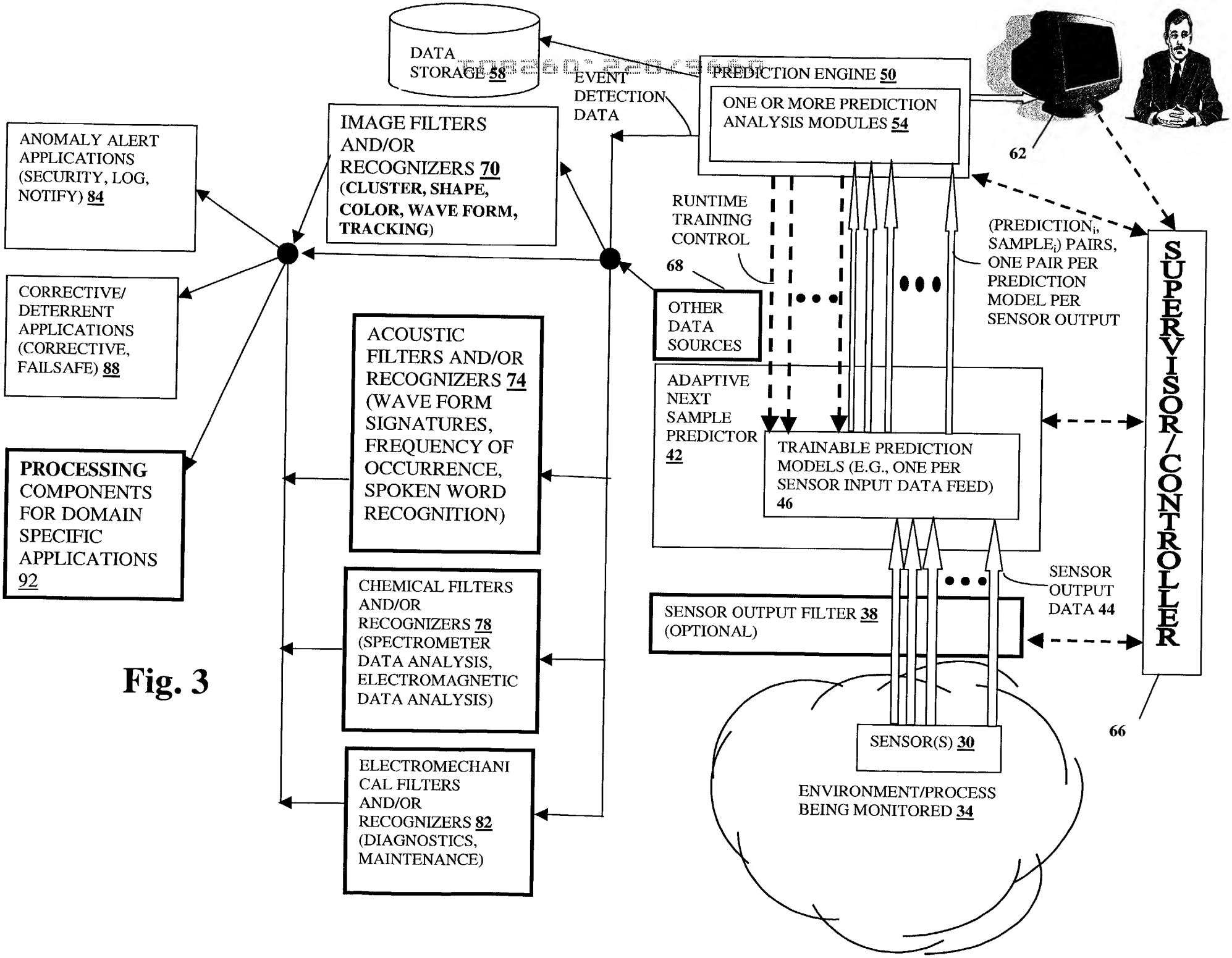

DATA STORAGE 58
EVENT DETECTION DATA
PREDICTION ENGINE 50
ONE OR MORE PREDICTION ANALYSIS MODULES 54
62
ANOMALY ALERT APPLICATIONS (SECURITY, LOG, NOTIFY) 84
IMAGE FILTERS AND/OR RECOGNIZERS 70 (CLUSTER, SHAPE, COLOR, WAVE FORM, TRACKING)
RUNTIME TRAINING CONTROL
68
(PREDICTION$_i$, SAMPLE$_i$) PAIRS, ONE PAIR PER PREDICTION MODEL PER SENSOR OUTPUT
SUPERVISOR/CONTROLLER
CORRECTIVE/ DETERRENT APPLICATIONS (CORRECTIVE, FAILSAFE) 88
OTHER DATA SOURCES
ACOUSTIC FILTERS AND/OR RECOGNIZERS 74 (WAVE FORM SIGNATURES, FREQUENCY OF OCCURRENCE, SPOKEN WORD RECOGNITION)
ADAPTIVE NEXT SAMPLE PREDICTOR 42
TRAINABLE PREDICTION MODELS (E.G., ONE PER SENSOR INPUT DATA FEED) 46
PROCESSING COMPONENTS FOR DOMAIN SPECIFIC APPLICATIONS 92
SENSOR OUTPUT DATA 44
SENSOR OUTPUT FILTER 38 (OPTIONAL)
CHEMICAL FILTERS AND/OR RECOGNIZERS 78 (SPECTROMETER DATA ANALYSIS, ELECTROMAGNETIC DATA ANALYSIS)
66
SENSOR(S) 30
ELECTROMECHANICAL FILTERS AND/OR RECOGNIZERS 82 (DIAGNOSTICS, MAINTENANCE)
ENVIRONMENT/PROCESS BEING MONITORED 34

Fig. 3

TD8260 "230796"

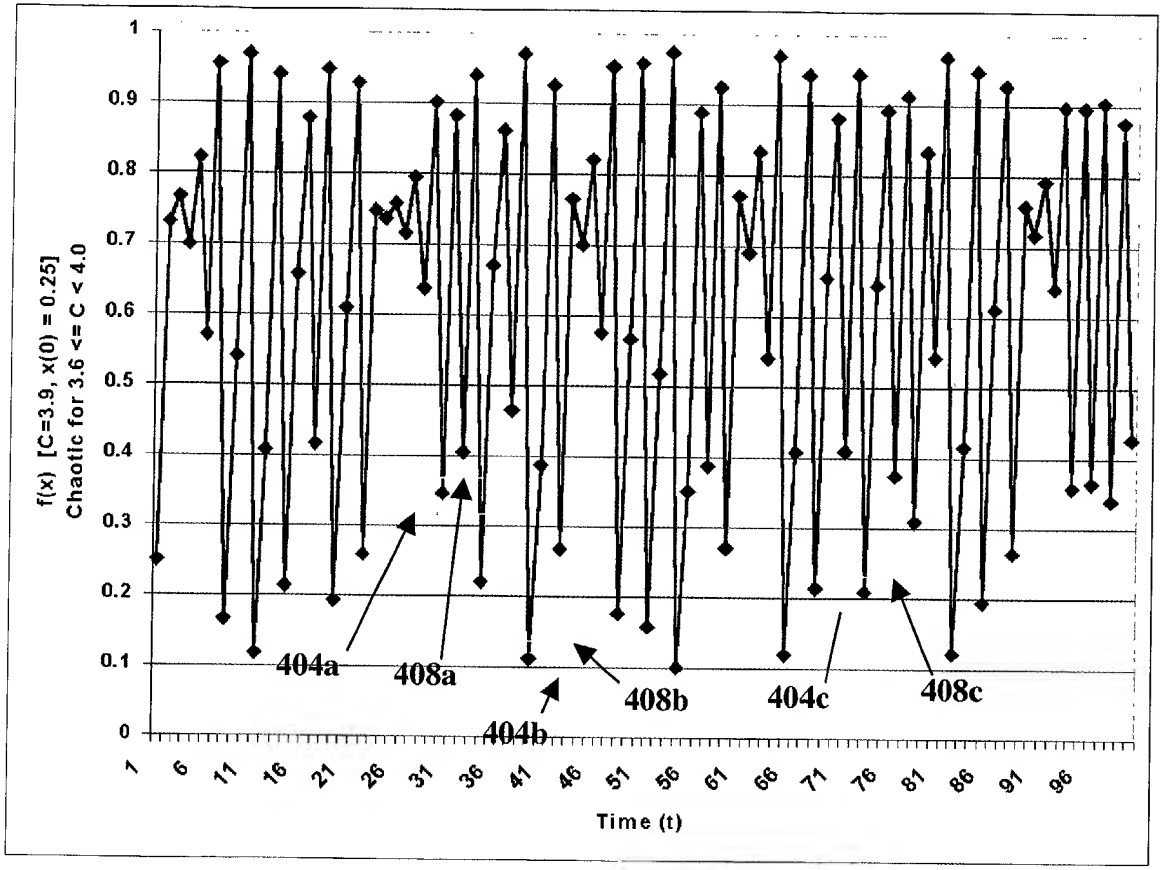


Fig. 4

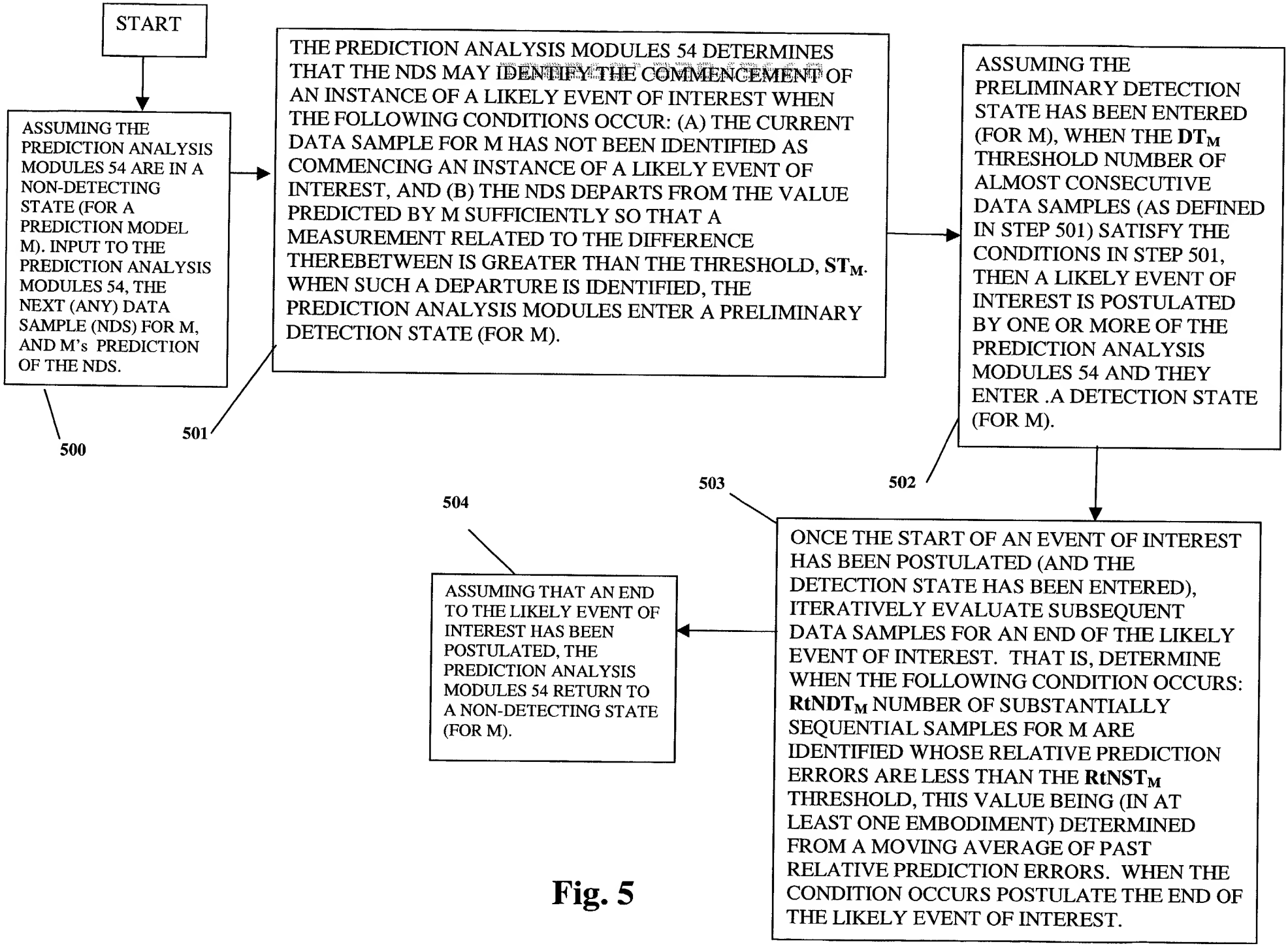
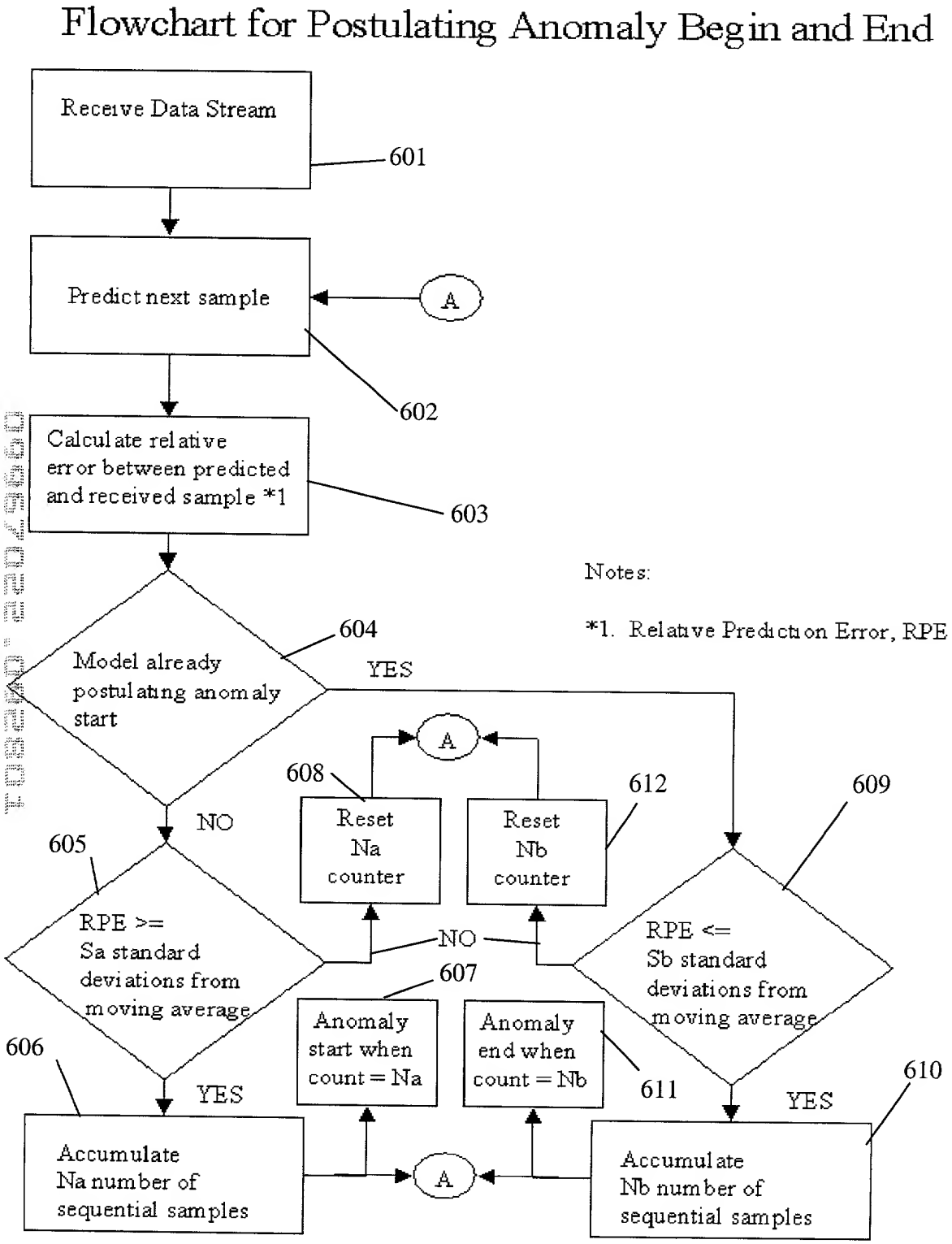

START
ASSUMING THE PREDICTION ANALYSIS MODULES 54 ARE IN A NON-DETECTING STATE (FOR A PREDICTION MODEL M). INPUT TO THE PREDICTION ANALYSIS MODULES 54, THE NEXT (ANY) DATA SAMPLE (NDS) FOR M, AND M's PREDICTION OF THE NDS.
500
THE PREDICTION ANALYSIS MODULES 54 DETERMINES THAT THE NDS MAY IDENTIFY THE COMMENCEMENT OF AN INSTANCE OF A LIKELY EVENT OF INTEREST WHEN THE FOLLOWING CONDITIONS OCCUR: (A) THE CURRENT DATA SAMPLE FOR M HAS NOT BEEN IDENTIFIED AS COMMENCING AN INSTANCE OF A LIKELY EVENT OF INTEREST, AND (B) THE NDS DEPARTS FROM THE VALUE PREDICTED BY M SUFFICIENTLY SO THAT A MEASUREMENT RELATED TO THE DIFFERENCE THEREBETWEEN IS GREATER THAN THE THRESHOLD, ST_M. WHEN SUCH A DEPARTURE IS IDENTIFIED, THE PREDICTION ANALYSIS MODULES ENTER A PRELIMINARY DETECTION STATE (FOR M).
501
ASSUMING THE PRELIMINARY DETECTION STATE HAS BEEN ENTERED (FOR M), WHEN THE DT_M THRESHOLD NUMBER OF ALMOST CONSECUTIVE DATA SAMPLES (AS DEFINED IN STEP 501) SATISFY THE CONDITIONS IN STEP 501, THEN A LIKELY EVENT OF INTEREST IS POSTULATED BY ONE OR MORE OF THE PREDICTION ANALYSIS MODULES 54 AND THEY ENTER .A DETECTION STATE (FOR M).
502
503
ONCE THE START OF AN EVENT OF INTEREST HAS BEEN POSTULATED (AND THE DETECTION STATE HAS BEEN ENTERED), ITERATIVELY EVALUATE SUBSEQUENT DATA SAMPLES FOR AN END OF THE LIKELY EVENT OF INTEREST. THAT IS, DETERMINE WHEN THE FOLLOWING CONDITION OCCURS: $RtNDT_M$ NUMBER OF SUBSTANTIALLY SEQUENTIAL SAMPLES FOR M ARE IDENTIFIED WHOSE RELATIVE PREDICTION ERRORS ARE LESS THAN THE $RtNST_M$ THRESHOLD, THIS VALUE BEING (IN AT LEAST ONE EMBODIMENT) DETERMINED FROM A MOVING AVERAGE OF PAST RELATIVE PREDICTION ERRORS. WHEN THE CONDITION OCCURS POSTULATE THE END OF THE LIKELY EVENT OF INTEREST.
504
ASSUMING THAT AN END TO THE LIKELY EVENT OF INTEREST HAS BEEN POSTULATED, THE PREDICTION ANALYSIS MODULES 54 RETURN TO A NON-DETECTING STATE (FOR M).

Fig. 5

Fig. 6


Flowchart for Postulating Anomaly Begin and End
Receive Data Stream
601
Predict next sample
A
602
Calculate relative error between predicted and received sample *1
603
Model already postulating anomaly start
604
YES
NO
605
RPE >= Sa standard deviations from moving average
606
YES
Accumulate Na number of sequential samples
A
608
Reset Na counter
A
612
Reset Nb counter
609
RPE <= Sb standard deviations from moving average
NO
607
Anomaly start when count = Na
Anomaly end when count = Nb
611
YES
610
Accumulate Nb number of sequential samples
Notes:
*1. Relative Prediction Error, RPE

T08360 "220796D

Fig. 7. Local and mean relative prediction error predicting a chaotic datastream

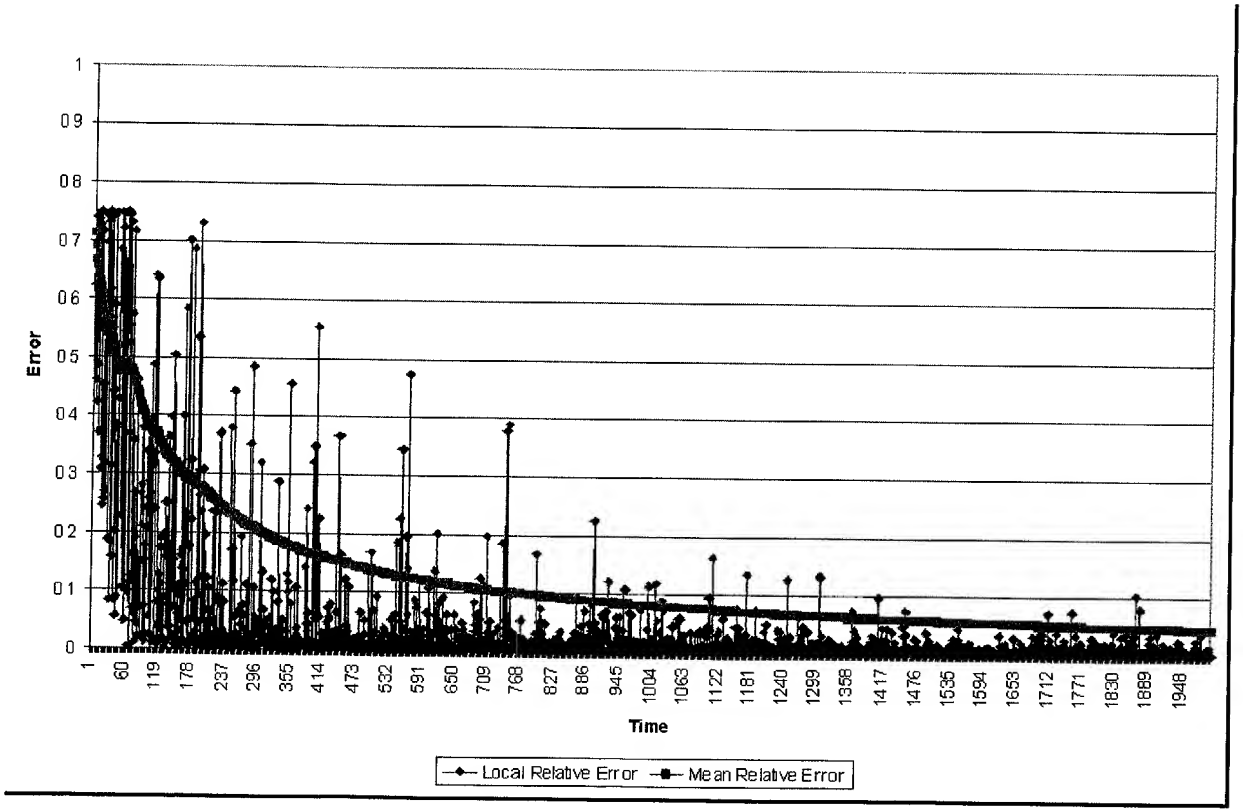
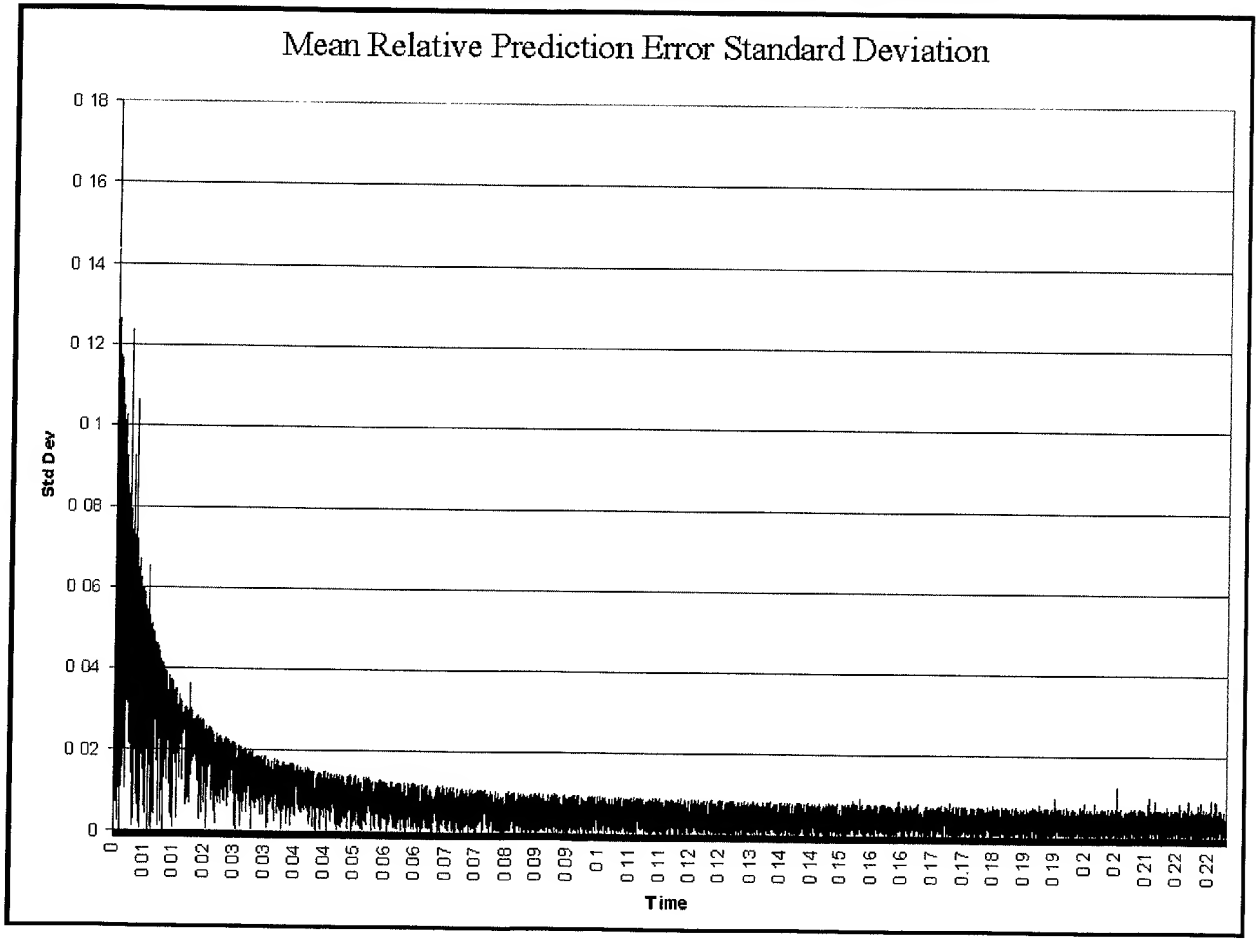


Fig. 8. Mean relative prediction error standard deviation



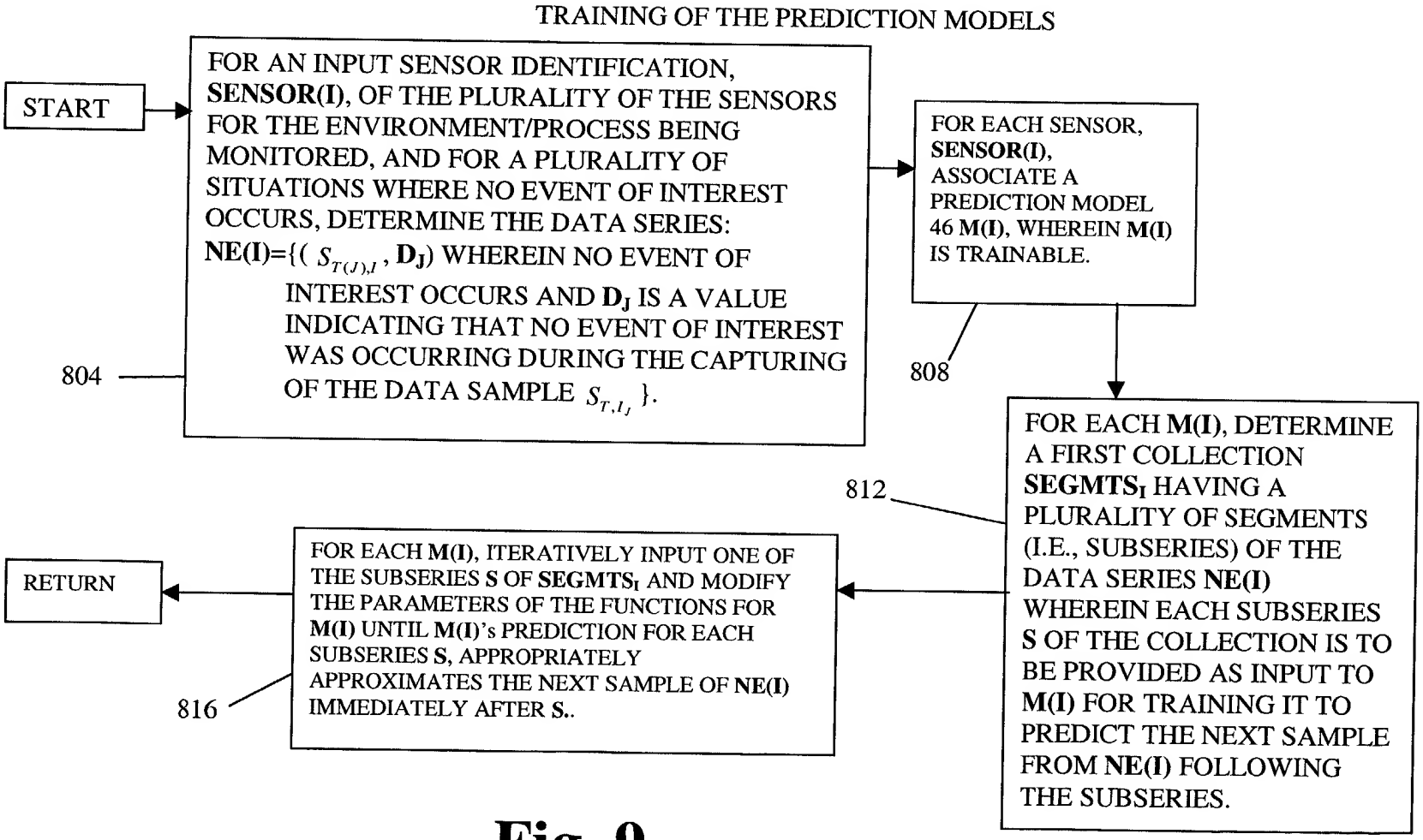

TRAINING OF THE PREDICTION MODELS
START
FOR AN INPUT SENSOR IDENTIFICATION, SENSOR(I), OF THE PLURALITY OF THE SENSORS FOR THE ENVIRONMENT/PROCESS BEING MONITORED, AND FOR A PLURALITY OF SITUATIONS WHERE NO EVENT OF INTEREST OCCURS, DETERMINE THE DATA SERIES: NE(I)={($S_{T(J),I}$, D_J) WHEREIN NO EVENT OF INTEREST OCCURS AND D_J IS A VALUE INDICATING THAT NO EVENT OF INTEREST WAS OCCURRING DURING THE CAPTURING OF THE DATA SAMPLE S_{T,I_J} }.
804
FOR EACH SENSOR, SENSOR(I), ASSOCIATE A PREDICTION MODEL 46 M(I), WHEREIN M(I) IS TRAINABLE.
808
FOR EACH M(I), DETERMINE A FIRST COLLECTION SEGMTS$_I$ HAVING A PLURALITY OF SEGMENTS (I.E., SUBSERIES) OF THE DATA SERIES NE(I) WHEREIN EACH SUBSERIES S OF THE COLLECTION IS TO BE PROVIDED AS INPUT TO M(I) FOR TRAINING IT TO PREDICT THE NEXT SAMPLE FROM NE(I) FOLLOWING THE SUBSERIES.
812
FOR EACH M(I), ITERATIVELY INPUT ONE OF THE SUBSERIES S OF SEGMTS$_I$ AND MODIFY THE PARAMETERS OF THE FUNCTIONS FOR M(I) UNTIL M(I)'s PREDICTION FOR EACH SUBSERIES S, APPROPRIATELY APPROXIMATES THE NEXT SAMPLE OF NE(I) IMMEDIATELY AFTER S..
816
RETURN

Fig. 9

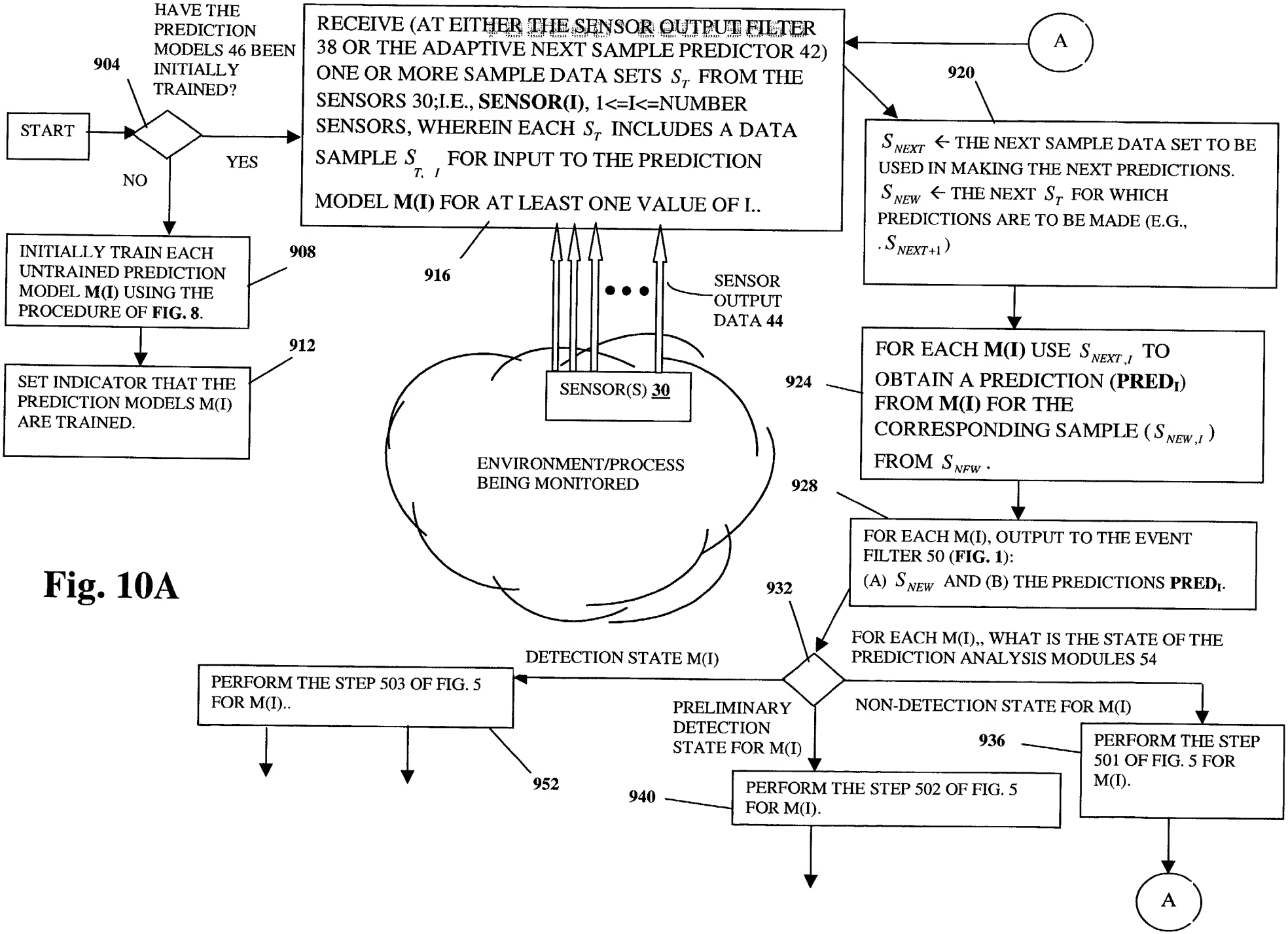

START
904
HAVE THE PREDICTION MODELS 46 BEEN INITIALLY TRAINED?
YES
NO
908
INITIALLY TRAIN EACH UNTRAINED PREDICTION MODEL M(I) USING THE PROCEDURE OF FIG. 8.
912
SET INDICATOR THAT THE PREDICTION MODELS M(I) ARE TRAINED.
RECEIVE (AT EITHER THE SENSOR OUTPUT FILTER 38 OR THE ADAPTIVE NEXT SAMPLE PREDICTOR 42) ONE OR MORE SAMPLE DATA SETS S_T FROM THE SENSORS 30;I.E., SENSOR(I), 1<=I<=NUMBER SENSORS, WHEREIN EACH S_T INCLUDES A DATA SAMPLE $S_{T,I}$ FOR INPUT TO THE PREDICTION MODEL M(I) FOR AT LEAST ONE VALUE OF I..
916
SENSOR OUTPUT DATA 44
SENSOR(S) 30
ENVIRONMENT/PROCESS BEING MONITORED
A
920
S_{NEXT} ← THE NEXT SAMPLE DATA SET TO BE USED IN MAKING THE NEXT PREDICTIONS.
S_{NEW} ← THE NEXT S_T FOR WHICH PREDICTIONS ARE TO BE MADE (E.G., S_{NEXT+1})
924
FOR EACH M(I) USE $S_{NEXT,I}$ TO OBTAIN A PREDICTION ($PRED_I$) FROM M(I) FOR THE CORRESPONDING SAMPLE ($S_{NEW,I}$) FROM S_{NFW}.
928
FOR EACH M(I), OUTPUT TO THE EVENT FILTER 50 (FIG. 1):
(A) S_{NEW} AND (B) THE PREDICTIONS $PRED_I$.
932
FOR EACH M(I),, WHAT IS THE STATE OF THE PREDICTION ANALYSIS MODULES 54
DETECTION STATE M(I)
PERFORM THE STEP 503 OF FIG. 5 FOR M(I)..
952
PRELIMINARY DETECTION STATE FOR M(I)
940
PERFORM THE STEP 502 OF FIG. 5 FOR M(I).
NON-DETECTION STATE FOR M(I)
936
PERFORM THE STEP 501 OF FIG. 5 FOR M(I).
A

Fig. 10A

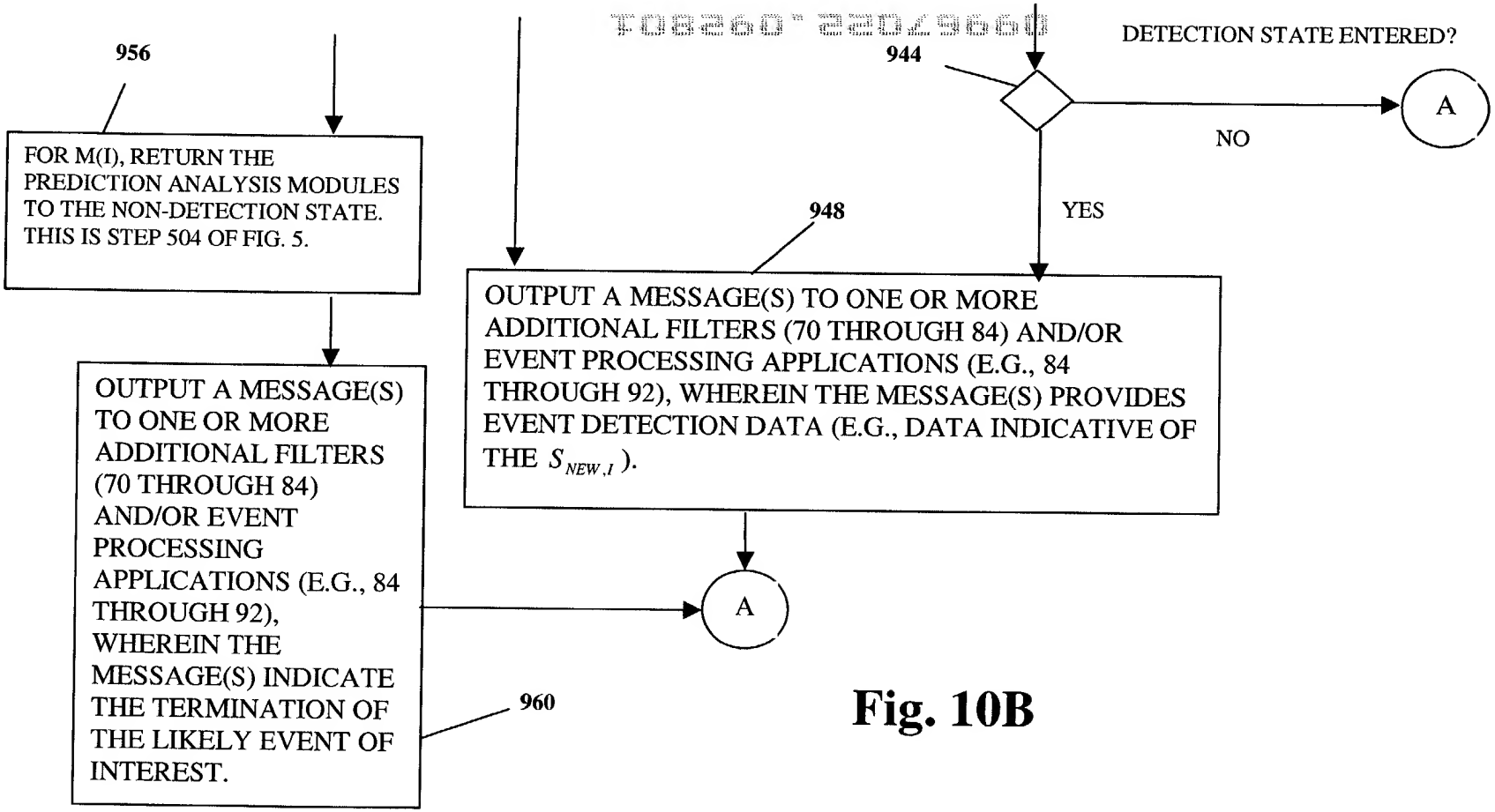

956
FOR M(I), RETURN THE PREDICTION ANALYSIS MODULES TO THE NON-DETECTION STATE. THIS IS STEP 504 OF FIG. 5.
OUTPUT A MESSAGE(S) TO ONE OR MORE ADDITIONAL FILTERS (70 THROUGH 84) AND/OR EVENT PROCESSING APPLICATIONS (E.G., 84 THROUGH 92), WHEREIN THE MESSAGE(S) INDICATE THE TERMINATION OF THE LIKELY EVENT OF INTEREST.
960
944
DETECTION STATE ENTERED?
NO
A
YES
948
OUTPUT A MESSAGE(S) TO ONE OR MORE ADDITIONAL FILTERS (70 THROUGH 84) AND/OR EVENT PROCESSING APPLICATIONS (E.G., 84 THROUGH 92), WHEREIN THE MESSAGE(S) PROVIDES EVENT DETECTION DATA (E.G., DATA INDICATIVE OF THE $S_{NEW,I}$).
A

Fig. 10B

Fig. 11

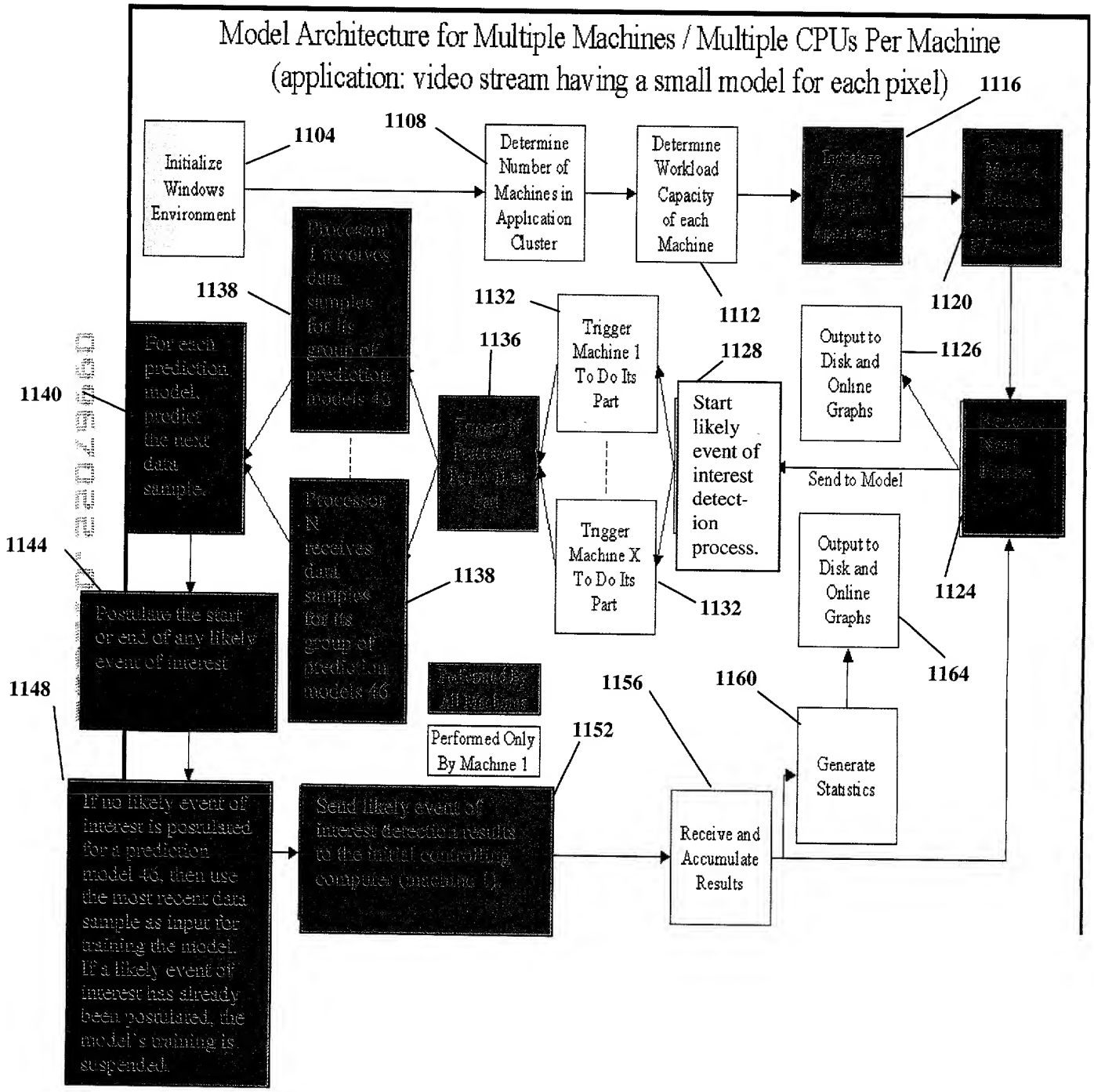
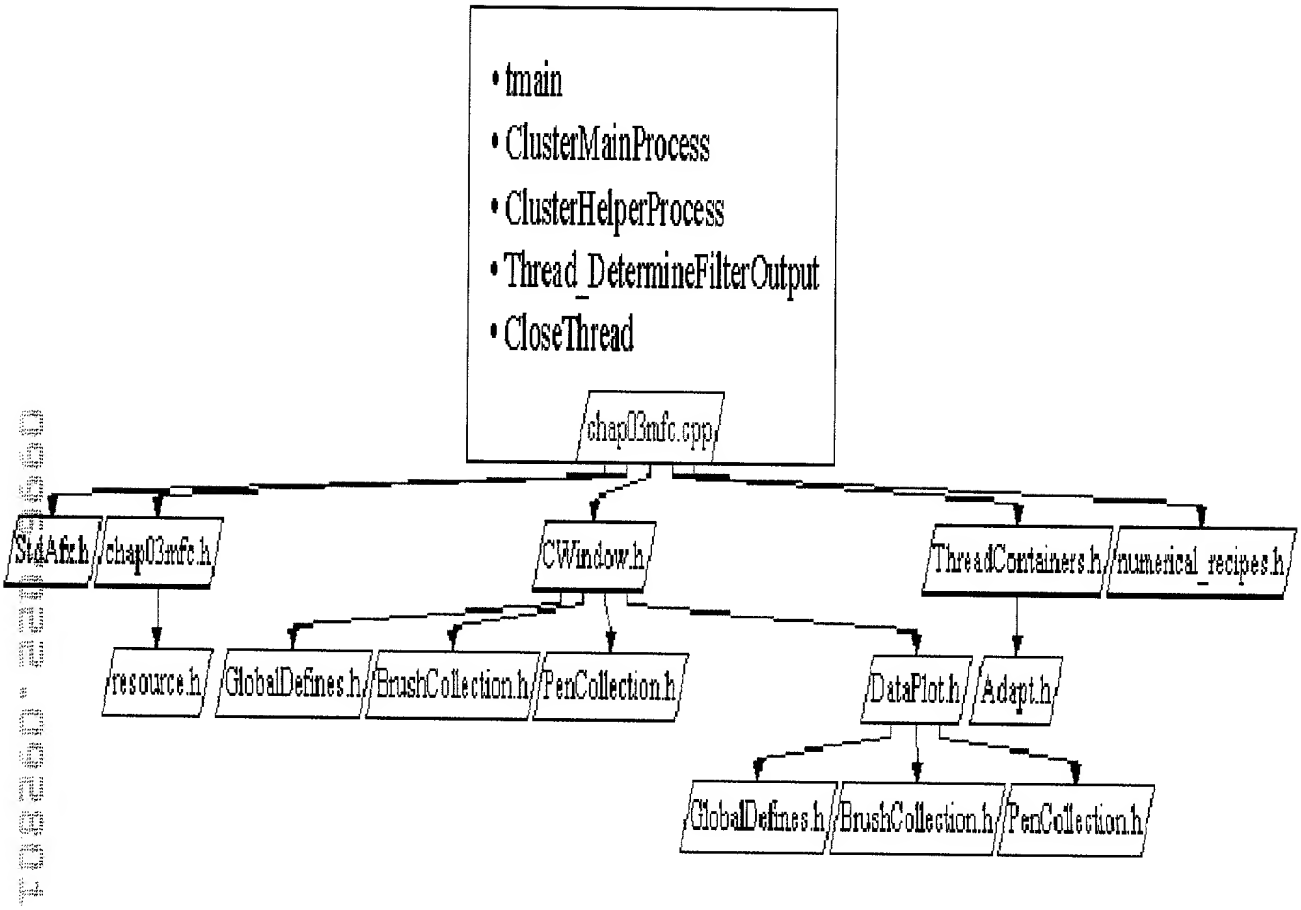

Model Architecture for Multiple Machines / Multiple CPUs Per Machine
(application: video stream having a small model for each pixel)
1116
Initialize Windows Environment
1104
1108
Determine Number of Machines in Application Cluster
Determine Workload Capacity of each Machine
1112
1120
1138
1132
1136
Trigger Machine 1 To Do Its Part
1128
Output to Disk and Online Graphs
1126
1140
For each prediction model, predict the next data sample.
Start likely event of interest detect-ion process.
Send to Model
Trigger Machine X To Do Its Part
1144
1138
1124
Output to Disk and Online Graphs
1164
Postulate the start or end of any likely event of interest
1148
1156
1160
Performed Only By Machine 1
1152
Generate Statistics
If no likely event of interest is postulated for a prediction model 46, then use the most recent data sample as input for training the model. If a likely event of interest has already been postulated, the model's training is suspended.
Receive and Accumulate Results

Fig. 12


• tmain
• ClusterMainProcess
• ClusterHelperProcess
• Thread_DetermineFilterOutput
• CloseThread
chap03mfc.cpp
StdAfx.h
chap03mfc.h
CWindow.h
ThreadContainers.h
numerical_recipes.h
resource.h
GlobalDefines.h
BrushCollection.h
PenCollection.h
DataPlot.h
Adapt.h
GlobalDefines.h
BrushCollection.h
PenCollection.h

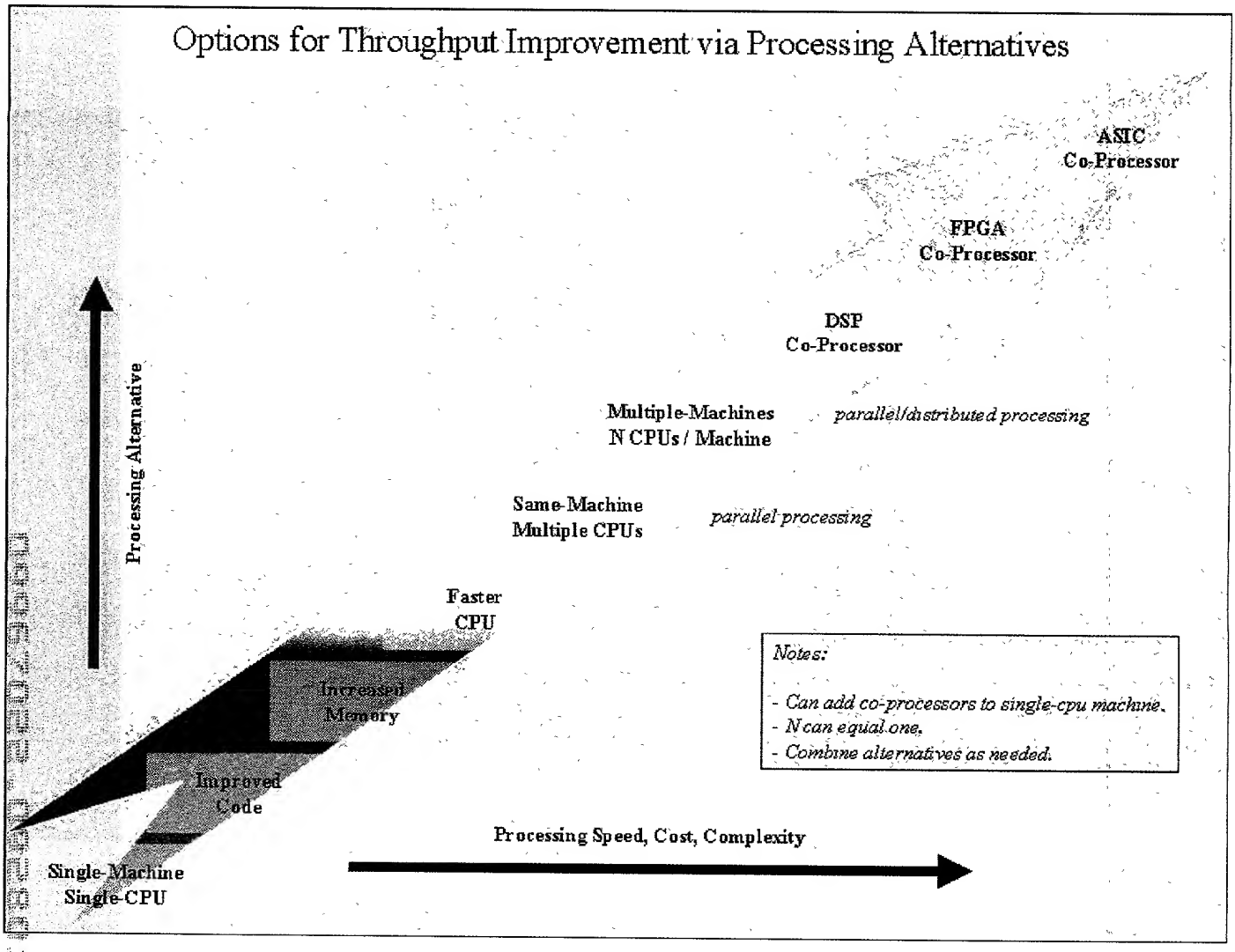

Options for Throughput Improvement via Processing Alternatives
ASIC
Co-Processor
FPGA
Co-Processor
DSP
Co-Processor
Multiple-Machines
N CPUs / Machine
parallel/distributed processing
Same-Machine
Multiple CPUs
parallel processing
Faster
CPU
Increased
Memory
Improved
Code
Single-Machine
Single-CPU
Processing Alternative
Processing Speed, Cost, Complexity
Notes:
- Can add co-processors to single-cpu machine.
- N can equal one.
- Combine alternatives as needed.

Fig. 13